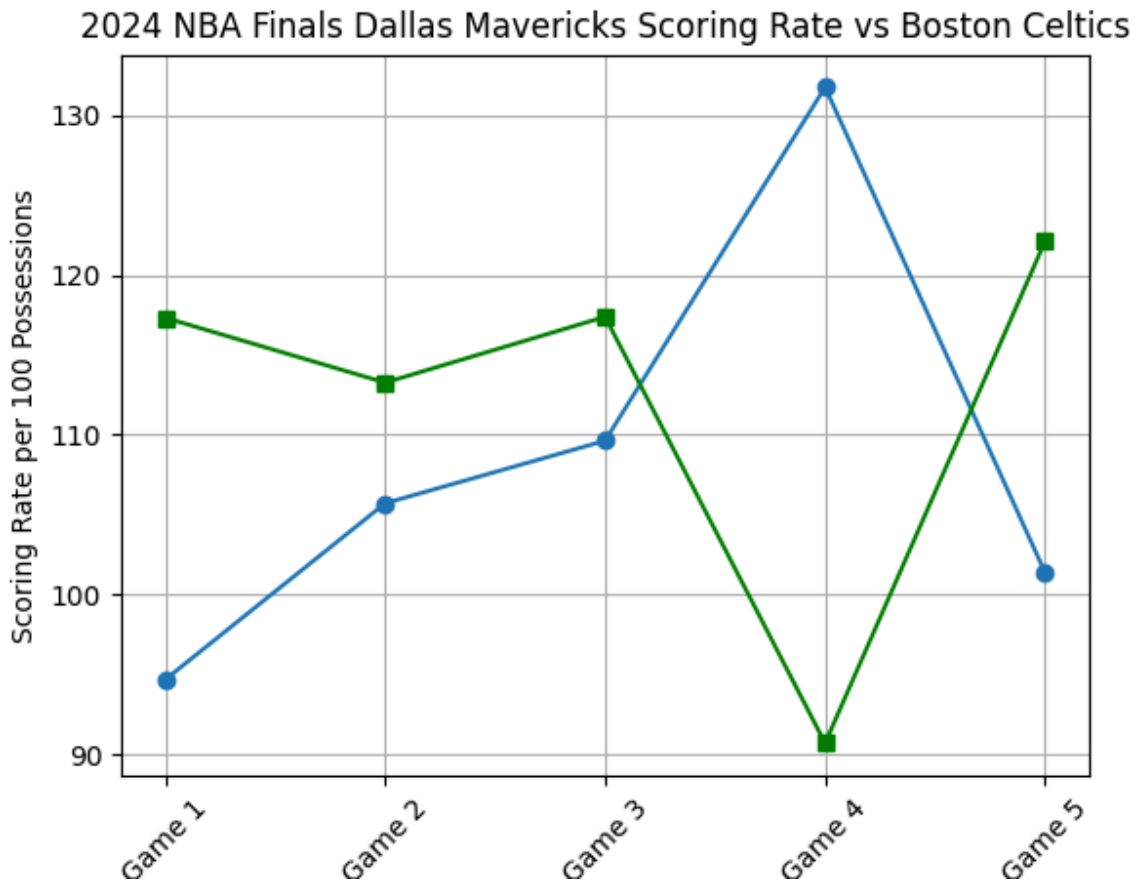
Bhargav Ashok - Compilation of Mini projects using Data Visualization tools with regards to Sports Data (I will add more as I go)

CELTICS VS MAV 2024 NBA FINALS DATA



Background knowledge about The 100 possession stat: This stat is meant to gauge team success the most accurately compared to plus-minus box player data, individual player stats, and separate categories like blocks per team or assists per player, etc. I was inspired by a wired auto-complete interview with a sports analyst which is on YouTube (just type in sports statistician answers questions), seems fairly simple but still cool to look at.

I was just messing around with some coding and got here. This plot specifically shows the data of the 2024 NBA finals; total points, per 100 possessions, per game between the Mavs and Celtics. I used Python Pandas and matplotlib to help me get started with the coding frameworks. Haven't worked with pandas much so I thought it would be cool to try it. I also used https://www.statmuse.com/ to get the possession stats for each game. Once again, it's a very simple graph but still wanted to showcase it here.

CODE:

```
import pandas as pd

# Sample data: list of dictionaries with game information
data = [
    {'game': 'Game 1', 'possessions': 94, 'scores': 89},
    {'game': 'Game 2', 'possessions': 92.7, 'scores': 98},
    {'game': 'Game 3', 'possessions': 90.3, 'scores': 99},
    {'game': 'Game 4', 'possessions': 92.6, 'scores': 122},
    {'game': 'Game 5', 'possessions': 86.8, 'scores': 88},

    ]

data_2 = [
    {'game_2': 'Game 1', 'possessions_2': 91.2, 'scores_2': 107},
    {'game_2': 'Game 2', 'possessions_2': 92.7, 'scores_2': 105},
    {'game_2': 'Game 3', 'possessions_2': 90.3, 'scores_2': 106},
    {'game_2': 'Game 4', 'possessions_2': 92.6, 'scores_2': 84},
    {'game_2': 'Game 5', 'possessions_2': 86.8, 'scores_2': 106},


    # Add more game data as needed
]

# Convert the data to a DataFrame for easier manipulation
df = pd.DataFrame(data)
df_2 = pd.DataFrame(data_2)

# Calculate the scoring rate per 100 possessions
df['scoring_rate_per_100_possessions'] = df['scores'] / df['possessions'] * 100
#celtics
df_2['scoring_rate_per_100_possessions_2'] = df_2['scores_2'] /
df_2['possessions_2'] * 100

# Analyze the data for mavericks
average_scoring_rate = df['scoring_rate_per_100_possessions'].mean()
max_scoring_rate = df['scoring_rate_per_100_possessions'].max()
min_scoring_rate = df['scoring_rate_per_100_possessions'].min()

# Analyze the data for Celtics
average_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].mean()
max_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].max()
min_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].min()

# Display the results
```

```
print("Average Scoring Rate per 100 Possessions:", average_scoring_rate)
print("Max Scoring Rate per 100 Possessions:", max_scoring_rate)
print("Min Scoring Rate per 100 Possessions:", min_scoring_rate)

# Display the results for Celtics
print("Average Scoring Rate per 100 Possessions_2:", average_scoring_rate_2)
print("Max Scoring Rate per 100 Possessions_2:", max_scoring_rate_2)
print("Min Scoring Rate per 100 Possessions_2:", min_scoring_rate_2)

# Optional: Plot the scoring rate for each game
import matplotlib.pyplot as plt

plt.plot(df['game'], df['scoring_rate_per_100_possessions'], marker='o')
#circle mark on graph
plt.plot(df_2['game_2'], df_2['scoring_rate_per_100_possessions_2'],
marker='s', color ="green") #square mark on graph
plt.title('2024 NBA Finals Dallas Mavericks Scoring Rate vs Boston Celtics')
plt.xlabel('Game')
plt.ylabel('Scoring Rate per 100 Possessions')
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```

HAWKS PRESENTATION:

The stats: this graph shows the Hawks's defensive and scoring rate over the last 25 games of the 2023-2024 season using the 100 possession stat.

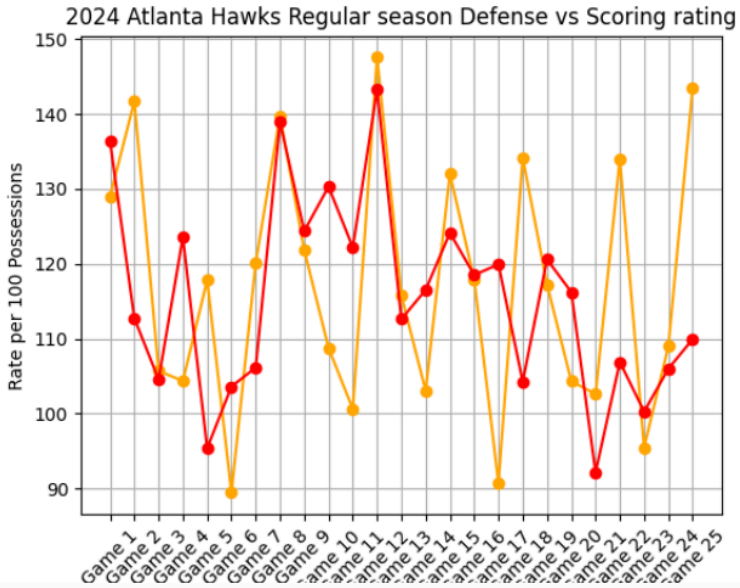
The line in orange represents the Defensive rating (also known as DRTG) where lower numbers indicate fewer possessions wasted/given up and correlate to better success (the Timberwolves had the best average DRTG rating of 109 points allowed per 100 possessions).

The line in red represents the scoring rating (also known as ORTG or simply points) where higher numbers indicate more points and correlate to better success per 100 possessions (the Pacers had the most points on average this season with 123.3 per 100 possessions).

As you can see on the graph, the orange line is higher than the red line during several instances, which means the Hawks gave up more points compared to scoring them. THATS NOT GOOD!

Average Offensive Rating = (Points/Possessions) * 100

Average Defensive Rating = (DRTG/Possessions) * 100



Graph made from Python using pandas and matplotlib libraries, stats taken from StatMuse

Code used for data:

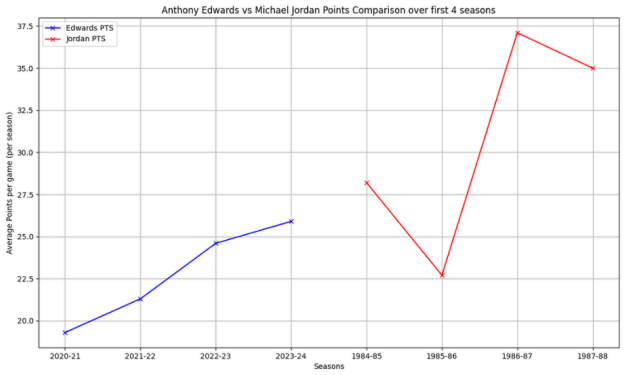
```
import pandas as pd

# Sample data: list of dictionaries with game information
data = [
    {'game': 'Game 1', 'possessions': 103.4, 'DRTG': 133.4},
    {'game': 'Game 2', 'possessions': 102.9, 'DRTG': 145.8},
    {'game': 'Game 3', 'possessions': 105.2, 'DRTG': 111.2},
    {'game': 'Game 4', 'possessions': 112.5, 'DRTG': 117.4},
    {'game': 'Game 5', 'possessions': 103.8, 'DRTG': 122.3},
    {'game': 'Game 6', 'possessions': 105.2, 'DRTG': 94.1},
    {'game': 'Game 7', 'possessions': 100.8, 'DRTG': 121.1},
    {'game': 'Game 8', 'possessions': 102.9, 'DRTG': 143.8},
    {'game': 'Game 9', 'possessions': 101.3, 'DRTG': 123.4},
    {'game': 'Game 10', 'possessions': 105.9, 'DRTG': 115.2},
    {'game': 'Game 11', 'possessions': 115.4, 'DRTG': 116.1},
    {'game': 'Game 12', 'possessions': 100.5, 'DRTG': 148.3},
    {'game': 'Game 13', 'possessions': 103.9, 'DRTG': 120.3},
    {'game': 'Game 14', 'possessions': 104.7, 'DRTG': 107.9},
```

```
    {'game': 'Game 15', 'possessions': 101.5, 'DRTG': 134.0},
    {'game': 'Game 16', 'possessions': 102.1, 'DRTG': 120.4},
    {'game': 'Game 17', 'possessions': 103.4, 'DRTG': 93.8},
    {'game': 'Game 18', 'possessions': 100.7, 'DRTG': 135.1},
    {'game': 'Game 19', 'possessions': 102, 'DRTG': 119.6},
    {'game': 'Game 20', 'possessions': 104.1, 'DRTG': 108.6},
    {'game': 'Game 21', 'possessions': 103.1, 'DRTG': 105.8},
    {'game': 'Game 22', 'possessions': 102.9, 'DRTG': 137.9},
    {'game': 'Game 23', 'possessions': 110.7, 'DRTG': 105.7},
    {'game': 'Game 24', 'possessions': 100, 'DRTG': 109.0},
    {'game': 'Game 25', 'possessions': 104.6, 'DRTG': 150.1},

    ]
#the lower the DRTG score the better
data_2 = [
    {'game_2': 'Game 1', 'possessions_2': 103.4, 'scores_2': 141},
    {'game_2': 'Game 2', 'possessions_2': 102.9, 'scores_2': 116},
    {'game_2': 'Game 3', 'possessions_2': 105.2, 'scores_2': 110},
    {'game_2': 'Game 4', 'possessions_2': 112.5, 'scores_2': 139},
    {'game_2': 'Game 5', 'possessions_2': 103.8, 'scores_2': 99},
    {'game_2': 'Game 6', 'possessions_2': 105.2, 'scores_2': 109},
    {'game_2': 'Game 7', 'possessions_2': 100.8, 'scores_2': 107},
    {'game_2': 'Game 8', 'possessions_2': 102.9, 'scores_2': 143},
    {'game_2': 'Game 9', 'possessions_2': 101.3, 'scores_2': 126},
    {'game_2': 'Game 10', 'possessions_2': 105.9, 'scores_2': 138},
    {'game_2': 'Game 11', 'possessions_2': 115.4, 'scores_2': 141},
    {'game_2': 'Game 12', 'possessions_2': 100.5, 'scores_2': 144},
    {'game_2': 'Game 13', 'possessions_2': 103.9, 'scores_2': 117},
    {'game_2': 'Game 14', 'possessions_2': 104.7, 'scores_2': 122},
    {'game_2': 'Game 15', 'possessions_2': 101.5, 'scores_2': 126},
    {'game_2': 'Game 16', 'possessions_2': 102.1, 'scores_2': 121},
    {'game_2': 'Game 17', 'possessions_2': 103.4, 'scores_2': 124},
    {'game_2': 'Game 18', 'possessions_2': 100.7, 'scores_2': 105},
    {'game_2': 'Game 19', 'possessions_2': 102, 'scores_2': 123},
    {'game_2': 'Game 20', 'possessions_2': 104.1, 'scores_2': 121},
    {'game_2': 'Game 21', 'possessions_2': 103.1, 'scores_2': 95},
    {'game_2': 'Game 22', 'possessions_2': 102.9, 'scores_2': 110},
    {'game_2': 'Game 23', 'possessions_2': 110.7, 'scores_2': 111},
    {'game_2': 'Game 24', 'possessions_2': 100, 'scores_2': 106},
    {'game_2': 'Game 25', 'possessions_2': 104.6, 'scores_2': 115},


    # Add more game data as needed
]

# Convert the data to a DataFrame for easier manipulation
df = pd.DataFrame(data)
df_2 = pd.DataFrame(data_2)
```

```
# Calculate the defensive rate per 100 possessions
df['DRTG_rate_per_100_possessions'] = df['DRTG'] / df['possessions'] * 100
#scoring rate
df_2['scoring_rate_per_100_possessions_2'] = df_2['scores_2'] /
df_2['possessions_2'] * 100

# Analyze the data for defense
average_DRTG_rate = df['DRTG_rate_per_100_possessions'].mean()
max_DRTG_rate = df['DRTG_rate_per_100_possessions'].max()
min_DRTG_rate = df['DRTG_rate_per_100_possessions'].min()

# Analyze the data for hawks
average_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].mean()
max_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].max()
min_scoring_rate_2 = df_2['scoring_rate_per_100_possessions_2'].min()

# Display the results for defense
print("Average DRTG Rate per 100 Possessions:", average_DRTG_rate)
print("Max DRTG Rate per 100 Possessions:", max_DRTG_rate)
print("Min DRTG Rate per 100 Possessions:", min_DRTG_rate)

# Display the results for offense
print("Average Scoring Rate per 100 Possessions_2:", average_scoring_rate_2)
print("Max Scoring Rate per 100 Possessions_2:", max_scoring_rate_2)
print("Min Scoring Rate per 100 Possessions_2:", min_scoring_rate_2)

# Optional: Plot the scoring rate for each game
import matplotlib.pyplot as plt

plt.plot(df['game'], df['DRTG_rate_per_100_possessions'], marker='o', color =
"orange") #circle mark on graph
plt.plot(df_2['game_2'], df_2['scoring_rate_per_100_possessions_2'],
marker='o', color ="red")
plt.title('2024 Atlanta Hawks Regular season Defense vs Scoring rating')
plt.xlabel('Hawks Regular season Defense vs Scoring Rating')
plt.ylabel('Rate per 100 Possessions')
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```

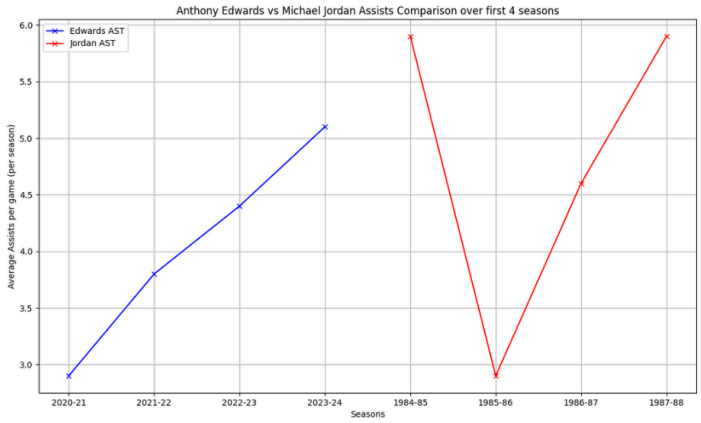
ANTHONY EDWARDS VS MICHAEL JORDAN GRAPHS

DO THE STATS GIVE US AN ANSWER (REGULAR SEASON POINTS)?



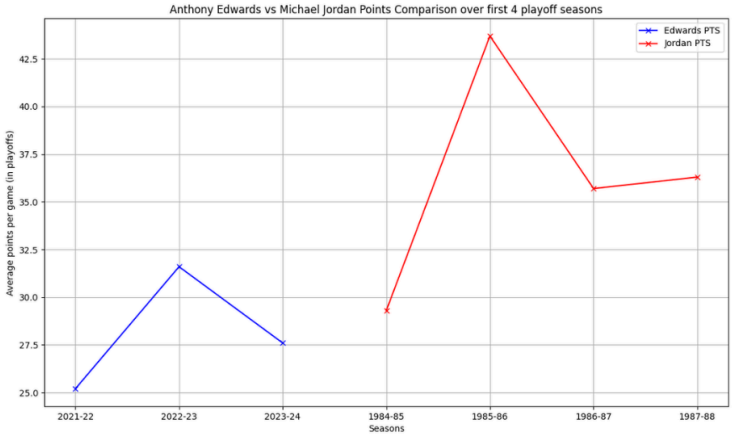
AS WE SEE HERE, JORDAN AVERAGED MORE POINTS IN HIS FIRST FOUR SEASONS COMPARED TO ANTHONY EDWARDS

DO THE STATS GIVE US AN ANSWER (REGULAR SEASON ASSISTS)?



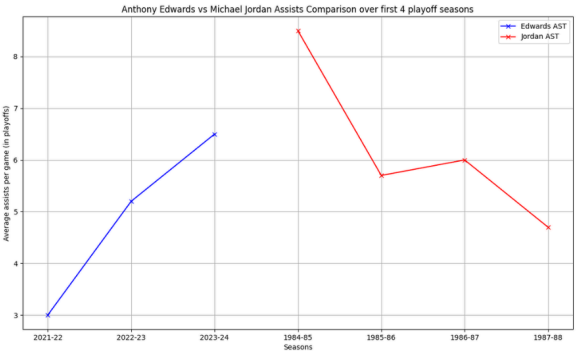
WHILE EDWARDS DID BETTER IN HIS SECOND SEASON COMPARED TO JORDAN, IT'S CLEAR THAT JORDAN HAD MORE ASSISTS TO BACK UP HIS GAME AS WELL

DO THE STATS GIVE US AN ANSWER (PLAYOFF POINTS EDITION)?



NOT LOOKING GOOD HERE FOR EDWARDS ON PAPER...

DO THE STATS GIVE US AN ANSWER (PLAYOFF ASSISTS EDITION)?



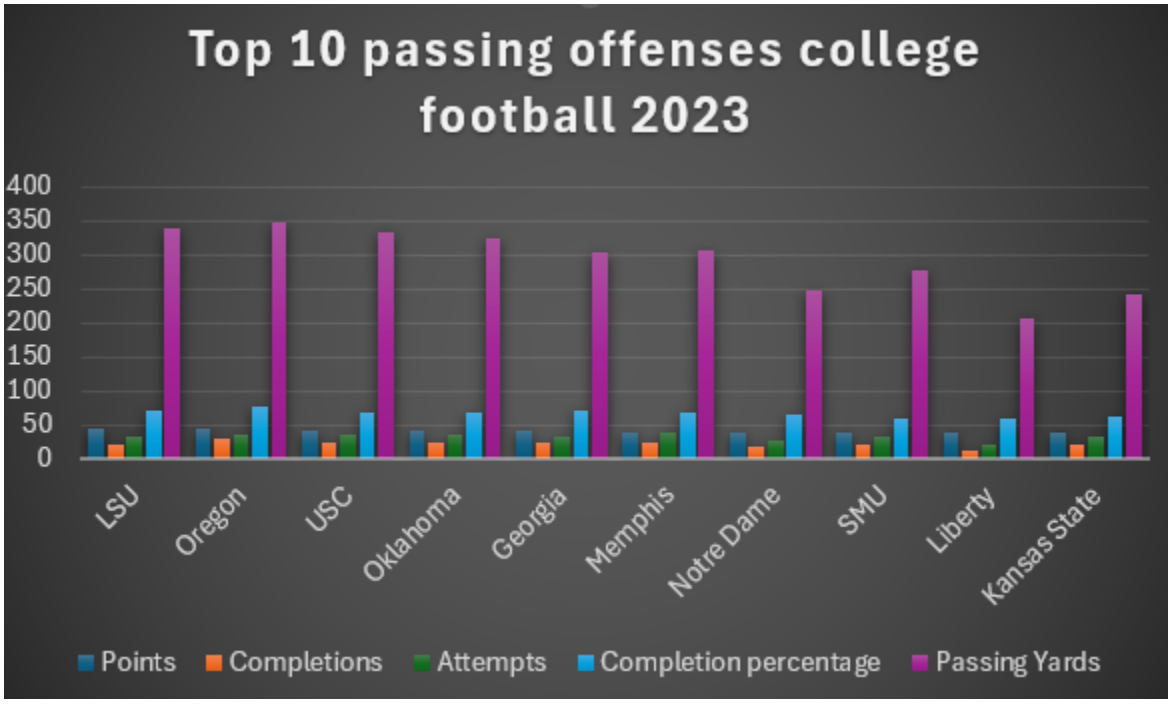
I MEAN... HE PERFORMED BETTER THAN JORDAN THIS YEAR COMPARED TO HIS LAST 2 SEASONS...

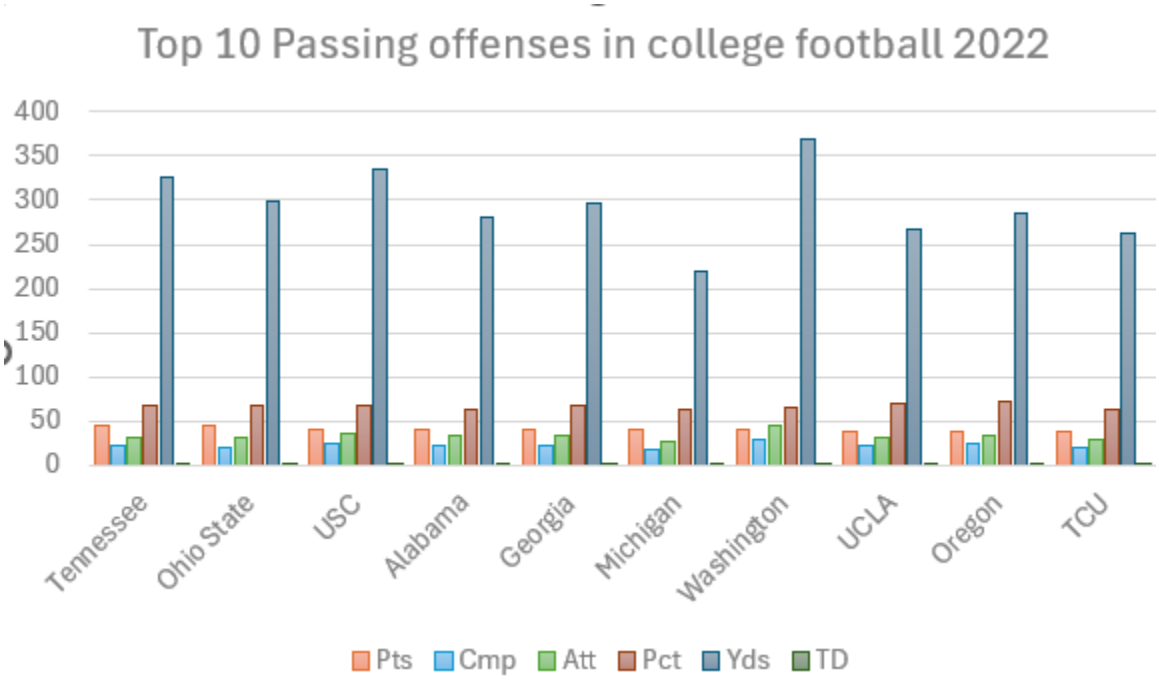
Part of code:

```
import pandas as pd
import matplotlib.pyplot as plt

# Load the data from the Excel files from basketball-reference
edwards_data = pd.read_excel('Anthony_Edwards_PLayoff_Stats.xlsx')
jordan_data = pd.read_excel('Michael_Jordan_Playoff_Stats.xlsx')

# Ensure the data contains stats up to age 22
edwards_data = edwards_data[edwards_data['Age'] <= 22] #first four NBA seasons
jordan_data = jordan_data[jordan_data['Age'] <= 24]

# Select relevant columns (for example: Points, Assists, Rebounds, etc.)
columns_of_interest = ['Season', 'AST']

edwards_stats = edwards_data[columns_of_interest]
jordan_stats = jordan_data[columns_of_interest]

# Plotting the stats comparison
plt.figure(figsize=(14, 8))

for column in columns_of_interest[1:]:
   plt.plot(edwards_stats['Season'], edwards_stats[column], marker='x', color =
"blue", label=f'Edwards {column}')
   plt.plot(jordan_stats['Season'], jordan_stats[column], marker='x', color =
"red", label=f'Jordan {column}')

plt.xlabel('Seasons')
plt.ylabel('Average assists per game (in playoffs)')
plt.title('Anthony Edwards vs Michael Jordan Assists Comparison over first 4
playoff seasons')
plt.legend()
plt.grid(True)
plt.show()
```

- Practiced Tableau Visualization with various NBA Datasets (Sacramento Kings, Hawks, etc.)
- Discussions about how to predict offense vs defensive stats in the NBA through my Condensed Media page

Excel graphs for predicting the best passing offense for college football





- Predicting best passing offense